CTF学习之MISC之图片隐写与文档隐写

原创

零安道长 于 2021-05-03 18:17:17 发布 745 收藏 16

分类专栏： CTF学习 文章标签： 网络安全

本文链接：https://blog.csdn.net/weixin_56977544/article/details/116379136

CTF学习之MISC之图片隐写

隐写术概述

- 图片隐写技术
- 图片EXIF信息隐写
- 图片LSB低位隐写

隐写术（Stega）

隐写术（Steganography，简写Stega）

```
一门关于信息隐藏的技巧与科学
信息隐藏：不让除预期的接收者之外的任何人知晓信息的传递事件或者信息的内容
与密码学(Cryptography）不同
```

隐写题目类型

- 图片隐写
  - 图片EXIF信息隐写
  - 图片高度缩减隐写
  - 图片LSB低位隐写
- 文档隐写
  - 文本隐写
  - 压缩包隐写
  - 其他文档格式（word、pdf、html文件）
  - 音频、视频隐写
  - 其他形式隐写

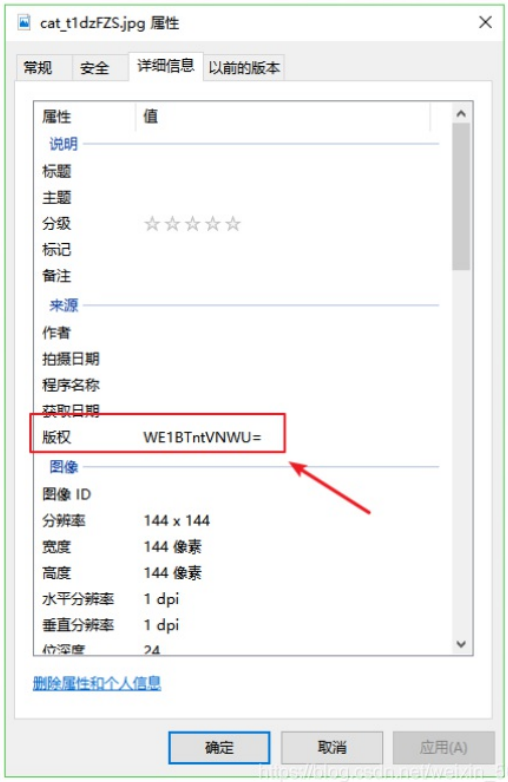
解题工具

- 图片隐写的解题工具
  - Strings，文本隐写分析工具
  - Binwalk，dd，foremost，图片分离工具
  - Winhex，ultraeditor，文件16 进制编辑器
  - Stegsolve.jar，图层分析工具-…
- 平台
  - kali系统自带strings，binwalk, dd, foremost等命令
  - windows平台下有Winhex，Stegsolve.jar等工具

图片EXIF信息隐写

- Exif（Exchangeable image file format）
  - 可交换图像文件格式。
  - 专门为数码相机的照片设定的，记录照片的属性信息和拍摄数据（曝光时间，拍摄时间，gps定位数据，相机品牌型号）。
- 支持的图片格式：JPEG、TIFF、XMP等等。
- 查看工具
  - 类Unix系统：exiftool
  - Windows系统：图片-右键-属性–详情

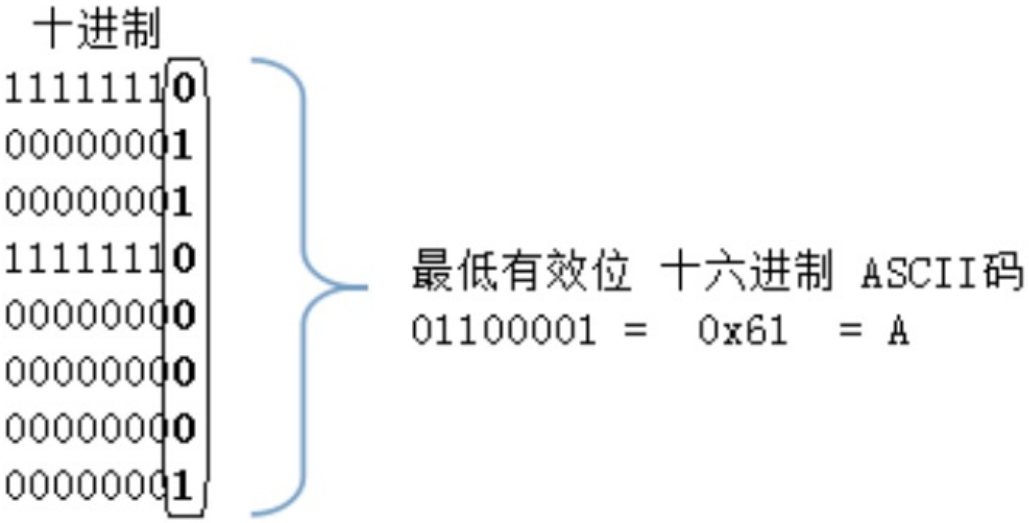
图片EXIF信息隐写



图片LSB隐写

- LSB（Least Significant Bit），最低有效位。
- 图片像素一般是由RGB三原色（即红绿蓝）组成的，每一种颜色占用8位，0x00~0xFF，即有256种值，一像素点共包含了256^3种的颜色；
- 人的肉眼能区分的只有其中一小部分；
- 修改RGB颜色分量中最低的二进制位，人眼无法区分。

- 例如：将字符“a”隐藏进图片LSB，即把“a”的二进制ascii码“01100001”，写到LSB通道的最低位。



解题思路

- 图片分析
- 图层分离，查看LSB

工具-可视化图层分析工具：

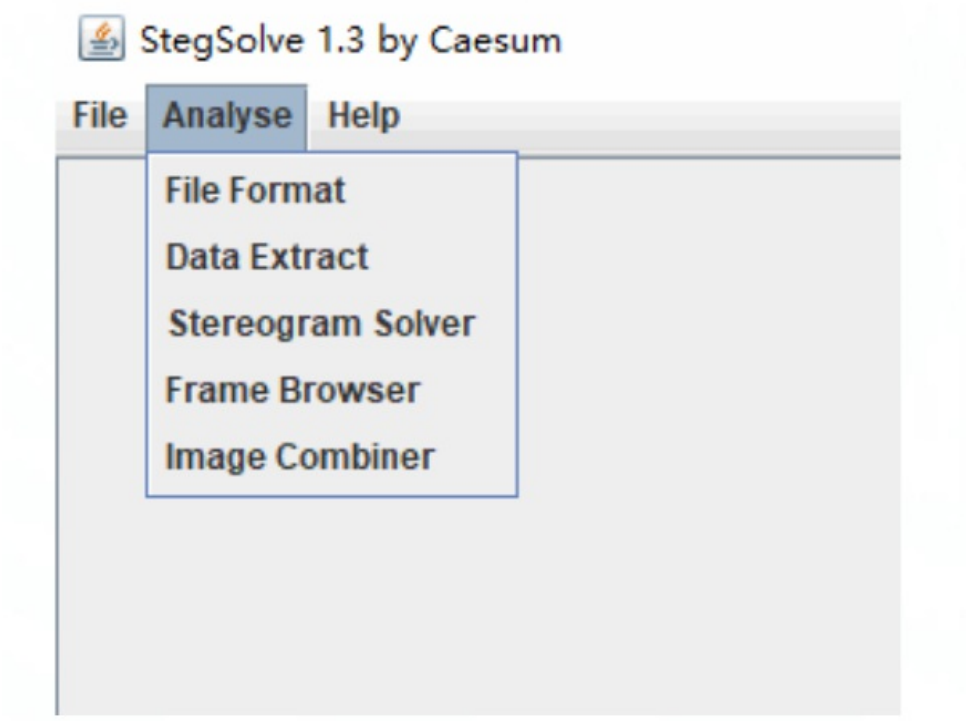
- Stegsolve.jar
- python脚本，PIL.Image库

解题工具：Stegsolve

- 需要java环境；
- 界面简单，协助分析图片图层、LSB信息。

主要功能：analyse

- File Format: 文件格式，查看图片的具体信息
- Data Extract: 数据抽取，抽取图片中隐藏数据
- Frame Browser: 帧浏览器，对GIF动图进行分离
- Image Combiner: 图片融合



图片隐写技术（进阶）

- 图片高度缩减隐写
- 图片结尾隐写

png图片格式

数据块符号	数据块名称	字节	16进制内容	描述
PNG signature	PNG签名	8	89 50 4E 47 0D 0A 1A 0A	标志着PNG图片的开始
IHDR	文件头数据块	13	-	图片的宽，高，位深，压缩方法等等，一个PNG数据流中只能有一个文件头数据块。
IDAT	图像数据块		-	存储图像实际的数据，可以存在多个数据块。
IEND	图像结束数据	12	00 00 00 00 49 45 4E 44 AE 42 60 82	用来标记PNG文件或者数据流已经结束

Hex	As Characters
89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52	.PNG.....IHDR
00 00 00 01 00 00 00 01 08 02 00 00 00 90 77 53	..............wS
DE 00 00 00 0E 49 44 41 54 78 DA 62 F8 CF C0 00	Þ....IDATxÚbøÏÀ.
10 60 00 03 01 01 00 66 FD 9F 24 00 00 00 00 49	.`.....fý.$....I
45 4E 44 AE 42 60 82	END®B`.

- png图片基本数据单元
  - png图片以数据块（chunk）为数据单元存储在计算机中。
  - 每个数据块包含以下4个部分：

名称	字节数	说明
Length(长度)	4字节	指定数据块中数据域的长度
Chunk Type Code(数据块类型码)	4字节	数据块类型码由ASCII字母(A-Z和a-z)组成
Chunk Data(数据块数据)	可变长度	存储按照Chunk Type Code指定的数据
CRC(循环冗余检测)	4字节	存储用来检测是否有错误的循环冗余码

- IHDR数据块

域的名称	字节数	说明
Width	4	图像宽度，以像素为单位
Height	4	图像高度，以像素为单位
Bit depth	1	图像深度： 索引彩色图像：1，2，4或8 灰度图像：1，2，4，8或16 真彩色图像：8或16
...	4	ColorType、Compression method、Filter method、Interlace method
CRC检验码	4	对IHDR的17字节进行crc32计算得到。

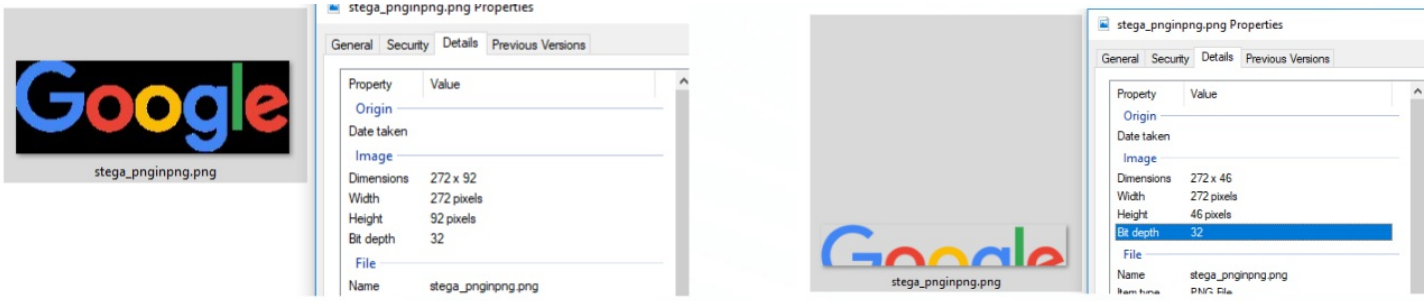
Hex	As Characters
89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52	.PNG....IHDR
00 00 00 01 00 00 00 01 08 02 00 00 00 90 77 53	..wS
DE 00 00 00 0E 49 44 41 54 78 DA 62 F8 CF C0 00	Þ....IDATxÚbøÏÀ.
10 60 00 03 01 01 00 66 FD 9F 24 00 00 00 00 49	.`.....fý.$....I
45 4E 44 AE 42 60 82	END®B`.

在线crc校验：http://www.ip33.com/crc.html

在线CRC校验：http://www.ip33.com/crc.html

图片高度缩减隐写

- 通过修改PNG图片的高度值，来对部分信息进行隐藏的。



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	%PNG IHDR
00000010	00	00	01	10	00	00	00	5C	08	06	00	00	00	A6	E7	EA	\ ¦çê
00000020	B6	00	00	17	18	49	44	41	54	78	01	ED	5D	0B	94	1C	¶ IDATx í] ”

- 解题思路
  - 分析题目提示；
  - 查看图片内容；
  - 计算校验和。（熟悉图片格式、crc校验、进制转换）

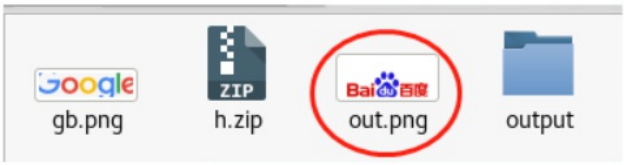
图片结尾隐写

- png图片结尾隐写
  - 在IEND数据块（49 45 4E 44 AE 42 60 82)之后附加额外数据内容，该数据不会被图片查看器加载、解析，从而达到隐藏信息的目的。
  - 图片结尾隐写解题工具
    - windows平台：Winhex，UltraEditor
    - Kali环境：binwalk，dd，foremost指令
  - 任何文件，都可以在文件尾附加隐写信息

Winhex的使用

- Winhex 实用技巧
  - 查找文本：Ctrl+F
  - 查找16进制：Alt+Ctrl+F-搜索下一个：F3
  - 快速选择大数据块：
    - 上下左右方向键移动光标
    - alt+1 选定要修改的数据块的起始字节；
    - 再使用alt+2选定结束字节；
  - 保存数据块到新文件：
    - 右键–edit –copy block –into new file

Binwalk分离图片




```
misc@mHost:~$ ls
gb.png  h.zip
misc@mHost:~$ binwalk gb.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 272 x 92, 8-bit/color RGBA, non-interlaced
5969          0x1751          PNG image, 540 x 258, 8-bit colormap, non-interlaced
6054          0x17A6          Zlib compressed data, best compression
9772          0x262C          Zlib compressed data, best compression

misc@mHost:~$ dd if=gb.png of=out.png skip=5969 bs=1
记录了7877+0 的读入
记录了7877+0 的写出
7877 bytes (7.9 kB, 7.7 KiB) copied, 0.00900255 s, 875 kB/s
misc@mHost:~$ ls
gb.png  h.zip  out.png
misc@mHost:~$ foremost gb.png
Processing: gb.png
|*|
```

Binwalk的使用

- Windows下binwalk的安装和使用
  - 下载
    - http://binwalk.org/
    - https://github.com/ReFirmLabs/binwalk
  - 安装
    - pip install setup.py / python setup.py install
  - 使用
    - python scripts\binwalk [filename]
    - 查找文本：Ctrl+F
    - 查找16进制：Alt+Ctrl+F
    - 搜索下一个：F3

文档结尾隐写

在文件的结尾标志符之后附加隐藏信息，而不破坏源文件

- 文本信息
- 其他文档

一般，隐写内容都是附加在源文件的结尾

- 如果附加在头部，破坏了文件头，可能导致文件无法识别\
- 如果附加在中间，有可能破坏了源文件的信息

常见文件格式的文件头、文件尾

文件格式	文件头标志位	文件尾标志位
JPEG (jpg)	FF D8 FF	FF D9
PNG (png)	89 50 4E 47	AE 42 60 82
GIF (gif)	47 49 46 38	00 3B
Archive (zip)	50 4B 03 04	50 4B

解题工具

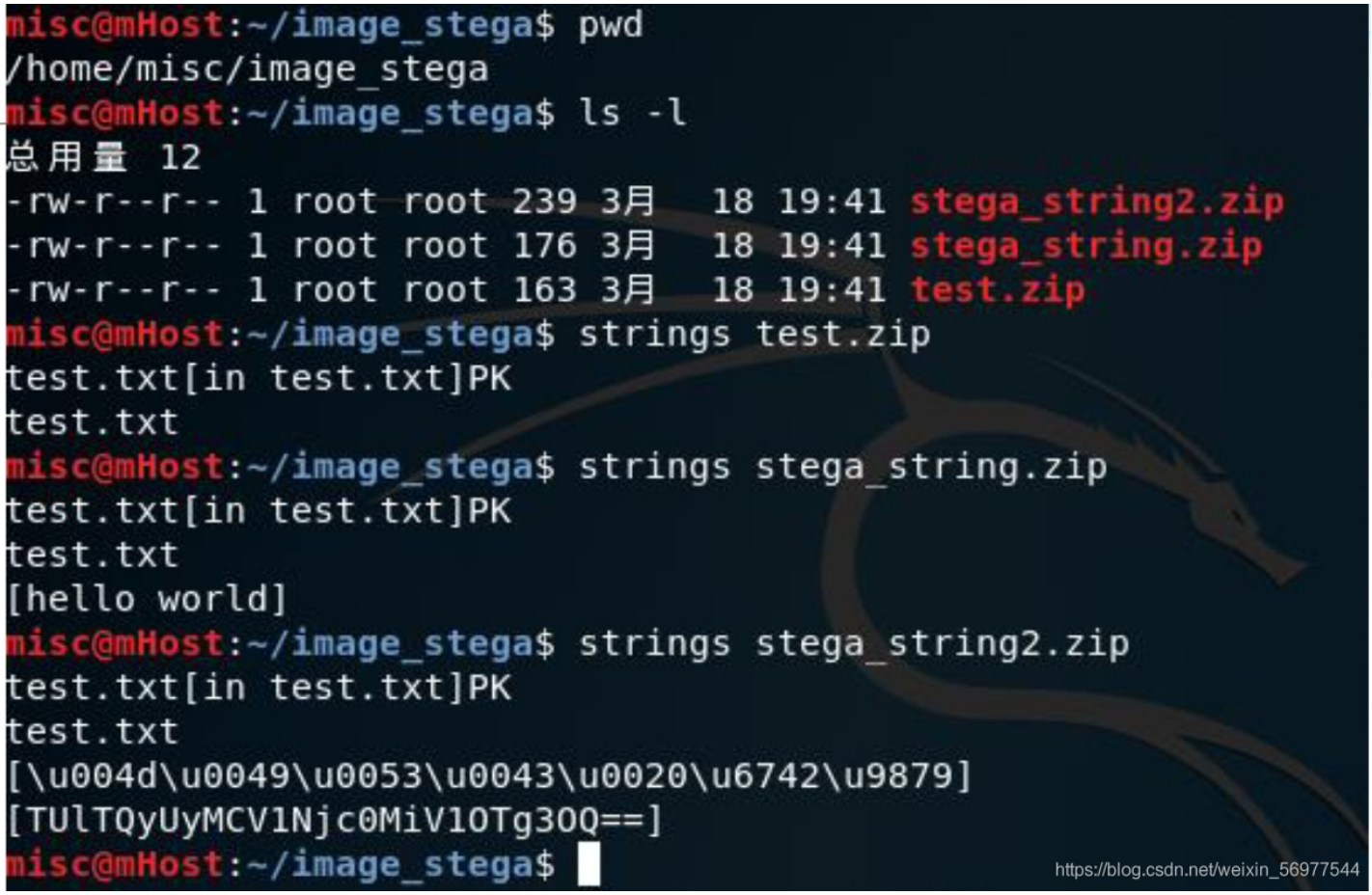
- strings命令：查找可打印字符串
- binwalk命令：分析文件隐写内容
- dd、foremost命令：分离文件

流程

- binwalk [filename] # 也可以添加“-e”参数直接分离
- foremost [filename] # 生成一个output的文件夹
- dd if=源文件.jpg of=目标文件.jpg skip=偏移量 bs=1

strings 用法

- strings命令
  - 在对象文件或二进制文件中查找可打印的字符串
- 选项
  - -a --all：扫描整个文件而不是只扫描目标文件初始化和装载段
  - -f –print-file-name：在显示字符串前先显示文件名-t --radix={o,d,x}：输出字符的位置，基于八进制，十进制或者十六进制
  - -e --encoding={s,S,b,l,B,L}：选择字符大小和排列顺序:s = 7-bit, S = 8-bit, {b,l} = 16-bit, {B,L} = 32-bit
- Tips-我们使用strings + [filename] 命令即可



更多资料

- 在线工具
  - 进制转换-在线工具 https://tool.lu/hexconvert/
  - CRC在线计算 http://www.ip33.com/crc.html
  - 在线工具——开源中国社区http://tool.oschina.net/
  - 工具大全（各种解密工具）http://tool.bugku.com/
- 练习平台：
  - i春秋，免费视频教程。https://www.ichunqiu.com/courses/ctf
  - 实验吧，在线实验环境、题库。http://www.shiyanbar.com/ctf/practice
  - 博客学习-CTF从入门到进（fang）阶(qi)之MISC -知乎 https://zhuanlan.zhihu.com/p/27598087
  - CTF入门指南(0基础) Angel_Kitty -博客园 https://www.cnblogs.com/ECJTUACM-873284962/p/6691817.html
  - Introduction | CTF Resources https://ctfs.github.io/resources/

参考于北理工大学慕课信息安全课程